DOCUMENT RESUME

ED 048 354 TM 000 430

AUTHOR	Roscoe, John T.; Kittleson, Howard M.
TITLE	A Modified Gauss-Jordan Procedure as an Alternative to Iterative Procedures in Multiple Regression.
PUB DATE	Feb 71
NOTE	24p.; Paper presented at the Annual Meeting of the American Educational Research Association, New York, New York, February 4-7, 1971
EDRS PRICE	EDRS Price MF-$0.65 HC-$3.29
DESCRIPTORS	*Algorithms, Correlation, *Multiple Regression Analysis, *Research Methodology, *Statistical Analysis
IDENTIFIERS	Fortran, *Gauss Jordan Procedure

ABSTRACT

Correlation matrices involving linear dependencies are common in educational research. In such matrices, there is no unique sclutiçn for the multiple regression coefficients. Although computer programs using iterative techniques are used to overcome this problem, these techniques possess certain disadvantages. Accordingly, a modified Gauss-Jordan technique, designed to function in the presence of linear dependencies is proposed. The modification is straightforward and incorporates a simple and efficient computer program. A comparative study, involving a series of experiments with both the modified Gauss-Jordan and iterative procedures is reported. Results indicate that the Gauss-Jordan modification performed best with simple problems and its primary advantages appear to be pedagogical. The complete computer program and examples of its output with various combinations of linear dependencies are included. (LR)

ED 048354

A MODIFIED GAUSS-JORDAN PROCEDURE AS AN ALTERNATIVE TO ITERATIVE PROCEDURES IN MULTIPLE REGRESSION[1]

John T. Roscoe and Howard M. Kittleson

Kansas State University

INTRODUCTION

Correlation matrices involving linear dependencies are a common occurence in educational research. Typically, these arise when:

(1) Standard scores which are a linear function of raw scores are included with the raw scores from which they were derived.

(2) Total scores which are the sum of two or more subtest scores are included with the subtest scores.

(3) Design matrices using binary variables to represent group membership sum to a unit vector.

(4) A variable is constant for the particular data involved in the analysis.

When the correlation matrix has one or more linear dependencies, there is no unique solution for the multiple regression coefficients. However, there is an infinity of solutions which satisfy the least squares criterion and which yield the identical multiple-R and R-square. Generally, a computational strategy which assigns a zero regression weight for each redundant variable will be preferred over one which assigns arbitrary weights even though the latter yields the least squares solution.

TM 000 430

---

[1]Paper prepared for presentation at the annual convention of the American Educational Research Association, February, 1971, New York City.

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
OFFICE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL OFFICE OF EDUCATION POSITION OR POLICY.

Traditional strategies for deriving the multiple regression coefficients (the Gauss-Jordan procedure, for example) do not ordinarily function in the presence of linear dependencies of the sort described. Typically, computer programs using these procedures cease computation and print an error statement when linear dependencies are encountered. This is the case with Gauss-Jordan programs reported by Cooley and Lohnes in Multivariate Procedures for the Behavioral Sciences and by Veldman in Fortran Programming for the Behavioral Sciences. Some programs first calculate the determinant of the correlation matrix; if the determinant is zero, there is a linear dependency, computation ceases, and an error statement is printed; if the determinant is non-zero, execution continues, and the solution is found using the Gauss-Jordan procedure.

The failure of a multiple regression program due to an unanticipated linear dependency can cause needless and often inconvenient delay in research data analysis. Of even greater importance to many investigators, the multiple regression approach to experimental design popularized by Bottenberg, Ward, and Jennings in A.E.R.A. pre-sessions calls for the conscious inclusion of linear dependencies. These men have found the solution to the problem in the use of iterative procedures which yield a solution meeting the least squares criterion despite the presence of linear dependencies. The multiple regression coefficients provided by these programs have an arbitrary appearance, and there is no provision for identification of redundant variables. Computer programs using iterative techniques to derive the regression coefficients

from the correlation matrix are reported in the Veldman text cited earlier and in Research Design in the Behavioral Sciences: Multiple Regression Approach by Kelly, et. al.

The Gauss-Jordan procedure provides a direct solution characterized by a specific process carried out once, in contrast to iterative techniques in which a cycle of approximations is carried out repeatedly until a prescribed accuracy is achieved. The authors question the use of iterative procedures with graduate students in education for these reasons: (1) These procedures are so complicated that it is impossible to share their logic with the typical user, (2) The arbitrariness of the regression coefficients is a source of confusion, and (3) There is no provision for identifying redundant variables.

The authors have modified the Gauss-Jordan procedure to function in the presence of linear dependencies; the modified procedure is straightforward and marked by its simplicity. It has been incorporated into an extremely simple and efficient multiple regression program. The result is a computer strategy as versatile as the iterative techniques, the logic of which is comprehensible to the average graduate student in education, and the utility of which may be effectively mastered in a single class period. The program assigns a zero weight for each redundant variable, the rank of the correlation matrix is determined, and degrees of freedom may be empirically determined if desired. The program, its logic and use, are reported in this paper.

SOME MATRIX ALGEBRA BACKGROUND

Given a system of linear equations in standard form--with variables and coefficients on the left, constants on the right:

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = c_1$$

$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = c_2$$

$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = c_3$$

The task is to solve for the values of the variables, given the values of the coefficients and constants. We shall represent the solution by b_1, b_2 and b_3.

A series of algebraic operations is performed on the equations such that the equalities are maintained, but the equations are reduced to:

$$1x_1 + 0x_2 + 0x_3 = b_1$$

$$0x_1 + 1x_2 + 0x_3 = b_2$$

$$0x_1 + 0x_2 + 1x_3 = b_3$$

Referring to the original system of linear equations, these may be regarded as two column vectors, one on the left side of the equal sign, the other on the right:

$$\begin{bmatrix} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 \\ a_{31}x_1 + a_{32}x_1 + a_{33}x_3 \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

From matrix algebra, it is apparent that the vector on the left is the product of the matrix of the coefficients and a column vector of the variables:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

This may also be expressed in the notation of matrix algebra as:

$$AX = C \quad \underline{or} \quad X = A^{-1}C$$

Similarly, the solution equations given earlier may be expressed in matrix form as:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

And this may also be expressed in matrix algebra notation as:

$$IX = B \quad \underline{or} \quad X = B$$

In its simplest form, the Gauss-Jordan approach consists essentially of augmenting the coefficient matrix by the constant vector:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & c_1 \\ a_{21} & a_{22} & a_{23} & c_2 \\ a_{31} & a_{32} & a_{33} & c_3 \end{bmatrix}$$

Then, elementary row operations are performed on the augmented matrix until it is transformed into the identity matrix augmented by the solution vector:

$$\begin{bmatrix} 1 & 0 & 0 & b_1 \\ 0 & 1 & 0 & b_2 \\ 0 & 0 & 1 & b_3 \end{bmatrix}$$

The iterative approach differs from the Gauss-Jordan only in the method of reducing the augmented matrix. The method is extremely complex and will not be discussed here.

MULTIPLE REGRESSION IN MATRIX NOTATION

Let S represent the sum of products matrix for all independent variables, G a column vector of sums of products of the dependent variable with each of the independent variables, and B a column vector of the desired regression coefficients. For simplicity of computation, deviation scores are used; the regression coefficients are the same whether raw scores (X_i) or deviation scores ($x_i = X_i - M$) are used. The normal equations for the least squares solution may be expressed in matrix algebra notation by:

$$SB = G \qquad \underline{or} \qquad B = S^{-1}G$$

And the solution may be obtained as outlined in the previous section.

Suppose, however, that a correlation matrix is substituted for the sum of products matrix. Let R represent the correlation matrix of all independent variables, H a column vector of correlations of the dependent variable with each of the independent variables, and B a column vector of the desired regression coefficients. Then, using the notation of matrix algebra:

$$RB = H \qquad \underline{or} \qquad B = R^{-1}H$$

And the solution may be obtained as before. It should be noted that the regression coefficients produced in this manner (that is, using the correlation matrix) are for use with relative deviation scores ($z_i = x_i / \sigma$). These coefficients are sometimes referred to as beta-weights or standard-weights. They may be transformed to raw-score weights by multiplying

each by the ratio of the standard deviation of the dependent variable to that of the respective independent variable: B_j (σ_y / σ_j). Equations for deriving the regression constant, multiple-R, and R-square may be found in most statistics textbooks.

The calculation of the correlation matrix and standard weights as intermediate steps in deriving the raw-score regression coefficients can be a source of computational error; however, these are useful information, and matrix operations of the sort described are likely to introduce less computational error with the numbers characteristic of correlation matrices than with those encountered in sum of products matrices.

COMPUTER STRAGEGY FOR SOLVING FOR THE REGRESSION COEFFICIENTS

The Gauss-Jordan subroutine offered by Cooley and Lohnes requires about sixty Fortran statements. The iterative routines cited earlier are somewhat longer. It is possible, however, to produce a Gauss-Jordan strategy for solving regression coefficients from a correlation matrix with as few as ten Fortran statements:

```
      DO 400 I=1, NR
      BB = R(I,I)
      DO 310 J=1,NC
310   R(I,J) = R(I,J) / BB
      DO 400 L=1,NR
      IF (I.EQ.L) GO TO 400
      AA = -R(L,I)
      DO 340 J=1,NC
340   R(L,J) = R(L,J) + R(I,J) * AA
400   CONTINUE
```

Where R is the augmented matrix described earlier, NR the number of rows and NC the number of columns (NC = NR + 1). The solution appears in the extreme right column as before.

A special problem arises in the execution of the routine listed above if the value of BB is zero (which often is the case when linear dependencies are present). Division by zero is undefined in the typical Fortran compiler, and computation ceases. This obstacle may be overcome by the insertion of a single Fortran statement between the second and third Fortran statements above:

```
IF (ABS(BB).LT.0.0001) BB = 1.0
```

Now, a number of interesting phenomena occur when the program is modified in this fashion: (1) Execution occurs despite the presence of linear dependencies, (2) The least squares solution is achieved, (3) Redundant variables receive zero coefficients, and (4) The rank of the matrix is the sum of the elements in the principle diagonal.

THE COMPLETE MULTIPLE REGRESSION PROGRAM

A copy of the complete multiple regression program and examples of its output with various combinations of the four types of linear dependency described earlier will be found in the appendix.

The program, in the form presented, is suitable for use with IBM 360 computers having 32K or more of core; it will produce a correlation matrix as large as 70 x 70, as many as 50 prediction equations each using any combination of 26 or fewer variables, and as many F-statistics as desired. Provision is made for variable generation internal to the program.

Only the values of IR (card reader designation) and of IP (Printer designation) need be changed to make the program operational with any 360 series computer; other minor modifications may be needed to use

the program with other computers. Only the dimension statements need be changed to take advantage of greater storage where this is available. Many investigators will want to introduce extended precision, which has been omitted in this program designed for student use.

The use of the program is described as follows:

(1) The values of IR and IP must be made to match the local computer system, and the appropriate local systems cards must be added.

(2) Data are read on the 1 Format as floating point variables, and the 1 Format may be altered to match the data. At Kansas State University, the program is reproduced with cards of two colors, and the Format is inserted between the first half of the deck which is blue and the second half which is pink.

(3) Variable generation, if desired, may be introduced at the same point in the program as the 1 Format. Suppose, for example, that the data cards contain ten variables, and that the investigator wishes an eleventh variable which is the square of the fifth. He need only add the Fortran statement: X(11) = X(5)**2. There is no limit to the number of variables which may be generated, except that the total number of variables is not to exceed seventy.

(4) Following the program deck and the required systems cards, the first user card is a title card which may contain any information the investigator wishes to use as a header for the output.

(5) The second user card is the main control card, and it contains a series of integer fields of five columns width. The first field is for N (the number of data cards), the second for NVP (the number of variables on each data card), the third for NEQ (the

number of prediction equations desired), the fourth for NF (the number of F-statistics desired), and the fifth for NGEN (the number of variables to be generated by the program).

(6) Data cards matching the 1 Format are next. There must be N of these.

(7) Next, the user must insert a control card for each regression equation. These cards contain a series of 26 integer fields three columns wide. Give the total number of variables (including the criterion) in the regression equation in the first field, then list the desired independent variables by number, finally give the number of the criterion. There must be NEQ of these control cards.

(8) Finally, the user must include a control card for each F-statistic desired. These cards contain four integer fields five columns wide. In the first field, give the number of the prediction equation to be used as a full model, then the number of the equation to be used as the restricted model, then the degrees of freedom for the numerator, and finally the degrees of freedom for the denominator. If the third and fourth fields are left blank, degrees of freedom will be determined empirically from the ranks of the matrices. The remainder of the F-statistic control card may be used to label the hypothesis test. There must be NF of these cards. R-square #50 has been set equal to zero in the program and may be used as the restricted model for the hypothesis of zero relationship. For a discussion of this and other hypotheses

which may be tested by comparing two regression models, the reader should consult Fundamental Research Statistics for the Behavioral Sciences or one of the other sources listed in the bibliography.

COMPARATIVE STUDY OF THE GAUSS-JORDAN AND ITERATIVE PROGRAMS

A series of simple experiments was conducted to compare the speed and accuracy of the program using the modified Gauss-Jordan procedure to that of a series of subroutines (adapted from Veldman) using an iterative procedure. It was necessary to add a coding feature to the iterative program (there is one in the Gauss-Jordan program reported here) to overcome excessive rounding error characteristic of certain data used in the experiments.

Both programs use an arbitrary internal criterion which influences the way they function. The arbitrary definition of zero (ABS(BB).LT.0.0001) in the Gauss-Jordan provides four decimal accuracy for R-square and the regression coefficients. The rank of the matrix with certain continuous data is a function of this criterion, and some investigators might want to reduce it by shifting the decimal one or two places to the left. The iterative program used a criterion of 0.00001 (difference between two successive R-squares) to stop iteration. With this stop criterion, it proved to be slightly less accurate (occasionally one or two digits at the fourth decimal place) than the Gauss-Jordan.

The Gauss-Jordan compiled in a third the time required for the iterative program, and it executed in about two-thirds the time with simple problems of the sort commonly reported as examples in statistics textbooks. However, with several more complicated examples, the

iterative program executed in about two-thirds the time required by the Gauss-Jordan. The primary advantages of the modified Gauss-Jordan appear to be pedagogical.

BIBLIOGRAPHY

Bottenberg, R. A., and J. H. Ward, Applied Multiple Linear Regression, U. S. Department of Commerce Office of Technical Services, AD413128, 1963.

Cooley, William W., and Paul R. Lohnes, Multivariate Procedures for the Behavioral Sciences, New York: John Wiley & Sons, Inc., 1962.

Fuller, Leonard E., Basic Matrix Theory, Englewood Cliffs: Prentice-Hall, Inc., 1962.

Jennings, Earl, "Fixed Effects Analysis of Variance by Regression Analysis," Multivariate Behavioral Research, January, 1967, pp. 95-107.

Jennings, Earl, and Joe H. Ward, "CDC 1604 Program for Intercorrelation and Multiple Regression Analysis," Behavioral Science, 1963, p. 81.

Kelly, Francis J., et. al., Research Design in the Behavioral Sciences: Multiple Regression Approach, Carbondale: Southern Illinois University Press, 1969.

Roscoe, John T., Fundamental Research Statistics for the Behavioral Sciences, New York: Holt, Rinehart and Winston, Inc., 1969.

Veldman, Donald J., Fortran Programming for the Behavioral Sciences, New York: Holt, Rinehart, and Winston, Inc., 1967.

Ward, Joe H., "Multiple Linear Regression Models," Computer Applications in the Behavioral Sciences, Borko, H. (Ed.). Englewood Cliffs: Prentice-Hall, Inc., 1962.

```
$JOB            JR,RUN=CHECK,TIME=15,PAGES=300
C MULTIPLE REGRESSION WITH VARIABLE GENERATION AND F-STATISTICS
C FIRST USER CARD IS TITLE CARD.
C 2ND USER CARD IS MAIN CONTROL IN I5 REPORTING N, NVD, NEQ, NF, NGEN
C    N = # DATA CARDS, NVD = # VARIABLES ON EACH DATA CARD, NEQ = #
C    EQUATIONS, NF = # F-STATISTICS, NGEN = # VARIABLES TO BE GENERAT
C N DATA CARDS FOLLOW IN 1 FORMAT.  ALL DATA IN FLOATING POINT MODE.
C NEQ CONTROL CARDS (ONE FOR EACH EQUATION) IN I3 FORMAT, REPORTING
C    TOTAL # VARIABLES IN EQUATION, LISTING OF VARIABLES, CRITERION V
C NF CONTROL CARDS IN I5 REPORTING FULL MODEL #, REST MODEL #, DF1, D
C    REMAINING SPACE ON F-STAT CARD MAY BE USED FOR LABEL OF HYPOTHES
 1       DIMENSION R(24,25), B(25), RSQ(50), IX(25), LABEL(15),  RANK(50
 2       DIMENSION TITLE(20), X(70), SUM(70), SS(70,70), STDV(70), C(70)
 3       IR = 1
 4       IP = 3
 5     3 FORMAT (26I3)
 6     4 FORMAT (4I5, 15A4)
 7     5 FORMAT (16I5)
 8     9 FORMAT (20A4)
 9    21 FORMAT ('1', '     TITLE:  ', 20A4, '  N =', I5//1X, 132('*'))
10    22 FORMAT (1X, 132('*') / 2X, 'VAR.NO.', 15(5X,I3) / 1X, 132('*'))
11    23 FORMAT (/5X, I3, 3X, 15F8.4)
12    24 FORMAT (/1X, 132('*')  / 3X, 'MEANS', 3X, 15F8.3)
13    25 FORMAT (/3X, 'STDV.', 3X, 15F8.3)
14    31 FORMAT(1X,132('*')/5X,'REGRESSION EQUATION NO.',I3,9X,
     -'CRITERION IS VAR. NO.',I3,25X,'RANK =',F5.0/1X,132('*')/
     = 10X,'VAR.NO.',5X,'B-WEIGHT',5X,'Z-WEIGHT')
15    32 FORMAT ( 15X, I3, 3X, F10.4, 3X, F10.4)
16    33 FORMAT (8X, 'CONSTANT =', 3X, F10.4//8X,'MULTPL-R =',3X,F10.4/
     - 8X, 'R-SQUARE =', 3X, F10.4/1X,132('*'))
17    51 FORMAT ( 5X, 'FULL MODEL', I3, 4X, 'RES. MODEL', I3, 4X,
     - 'F =', F9.3, 4X, 'DF =', I3, ',', I5,4X,15A4/1X,132('*'))
C READ TITLE AND MAIN CONTROL CARD, AND ZERO STORAGE.
18   100 READ (IR,9) TITLE
19       READ (IR,5) N, NVD, NEQ, NF, NGEN
20       RSQ(50) = 0.0
21       RANK(50) = 0.0
22       NV = NVD + NGEN
23       DO 110 J=1,NV
24       SUM (J) = 0.0
25       DO 110 I=1,NV
26   110 SS(I,J) = 0.0
C READ AND CODE DATA, ACCUMULATE SUMS AND RAW SUMS OF SQUARES.
27       DO 130 K=1,N
28       DO 120 J=NVD,NV
29   120 X(J) = 0.0
30       READ (IR,1) (X(J), J=1,NVD)
C 1 FORMAT AND VARIABLE GENERATION CARDS (IF ANY) FOLLOW THIS CARD.
31     1 FORMAT(20F2.0)
C 1 FORMAT AND VARIABLE GENERATION CARDS (IF ANY) PRECEDE THIS CARD.
32       IF (K.GT.1) GO TO 125
33       DO 122 J=1,NV
34   122 C(J) = X(J)
35   125 DO 126 J=1,NV
36       X(J) = X(J) - C(J)
37   126 SUM(J) = SUM(J) + X(J)
38       DO 130 I=1,NV
39       DO 130 J=1,NV
40   130 SS(I,J) = SS(I,J) + X(I) * X(J)
41   200 WRITE (IP,21) TITLE, N
```

```
,RUN=CHECK,TIME=15,PAGES=300
SION WITH VARIABLE GENERATION AND F-STATISTICS                    UU 0010
 IS TITLE CARD.                                                   UU 0020
S MAIN CONTROL IN I5 REPORTING N, NVD, NEQ, NF, NGEN.            UU 0030
ARDS, NVD = # VARIABLES ON EACH DATA CARD, NEQ = #               UU 0040
F = # F-STATISTICS, NGEN = # VARIABLES TO BE GENERATED.          UU 0050
LLOW IN 1 FORMAT.  ALL DATA IN FLOATING POINT MODE.              UU 0060
DS (ONE FOR EACH EQUATION) IN I3 FORMAT, REPORTING               UU 0070
ABLES IN EQUATION, LISTING OF VARIABLES, CRITERION LAST          UU 0080
S IN I5 REPORTING FULL MODEL #, REST MODEL #, DF1, DF2.          UU 0090
ACE ON F-STAT CARD MAY BE USED FOR LABEL OF HYPOTHESIS.          UU 0100
(24,25), B(25), RSQ(50), IX(25), LABEL(15),  RANK(50)            UU 0110
ITLE(20), X(70), SUM(70), SS(70,70), STDV(70), C(70)             UU 0120
                                                                 UU 0130
                                                                 UU 0140
3)                                                               UU 0150
, 15A4)                                                          UU 0160
5)                                                               UU 0170
4)                                                               UU 0180
', '     TITLE:  ', 20A4, '  N =', I5//1X, 132('*'))            UU 0190
 132('*') / 2X, 'VAR.NO.', 15(5X,I3) / 1X, 132('*'))            UU 0200
, I3, 3X, 15F8.4)                                                UU 0210
, 132('*')  / 3X, 'MEANS', 3X, 15F8.3)                           UU 0220
, 'STDV.', 3X, 15F8.3)                                           UU 0230
32('*')/5X,'REGRESSION EQUATION NO.',I3,9X,                      UU 0240
IS VAR. NO.',I3,25X,'RANK =',F5.0/1X,132('*')/                   UU 0250
0.',5X,'B-WEIGHT',5X,'Z-WEIGHT')                                 UU 0260
X, I3, 3X, F10.4, 3X, F10.4)                                     UU 0270
 'CONSTANT =', 3X, F10.4//8X,'MULTPL-R =',3X,F10.4/              UU 0280
ARE =', 3X, F10.4/1X,132('*'))                                   UU 0290
, 'FULL MODEL', I3, 4X, 'RES. MODEL', I3, 4X,                    UU 0300
3, 4X, 'DF =', I3, ',', I5,4X,15A4/1X,132('*'))                  UU 0310
MAIN CONTROL CARD, AND ZERO STORAGE.                             UU 0320
 TITLE                                                           UU 0330
 N, NVD, NEQ, NF, NGEN                                           UU 0340
.0                                                               UU 0350
C.0                                                              UU 0360
NGEN                                                             UU 0370
NV                                                               UU 0380
.0                                                               UU 0390
NV                                                               UU 0400
.0                                                               UU 0410
ATA, ACCUMULATE SUMS AND RAW SUMS OF SQUARES.                    UU 0420
N                                                                UU 0430
C,NV                                                             UU 0440
                                                                 UU 0450
 (X(J), J=1,NVD)                                                 UU 0460
RIABLE GENERATION CARDS (IF ANY) FOLLOW THIS CARD.               UU 0470
.0)                                                              UU 0480
RIABLE GENERATION CARDS (IF ANY) PRECEDE THIS CARD.              UU 0490
 GO TO 125                                                       UU 0500
NV                                                               UU 0510
                                                                 UU 0520
NV                                                               UU 0530
 - C(J)                                                          UU 0540
M(J) + X(J)                                                      UU 0550
NV                                                               UU 0560
NV                                                               UU 0570
S(I,J) + X(I) * X(J)                                             UU 0580
1) TITLE, N                                                      UU 0590
```

```
      C CALCULATE SUMS OF SQUARES, THEN MEANS AND STDVS, THEN CORREL
42          FN = N
43          DO 240 I = 1,NV
44          DO 240 J=1,NV
45        240 SS(I,J) = SS(I,J) - (SUM(I) * SUM(J) / FN)
46          DO 250 J=1,NV
47          SUM(J) = SUM(J) / FN + C(J)
48        250 STDV(J) = SQRT (SS(J,J) / FN) + 0.0000001
49          DO 260 I=1,NV
50          DO 260 J=1,NV
51        260 SS(I,J) = SS(I,J) / (FN * STDV(I) * STDV(J))
      C PRINT CORRELATIONS, MEANS, AND STANDARD DEVIATIONS IN MATRI
52          JJ = 1
53          MM = (NV - 1) / 15 + 1
54          DO 280 M=1,MM
55          JJJ = M * 15
56          WRITE (IP,22) (J, J=JJ,JJJ)
57          JJJ = JJ + 14
58          IF (NV.LE.JJJ) JJJ = NV
59          DO 270 I=1,NV
60        270 WRITE (IP,23) I, (SS(I,J), J=JJ,JJJ)
61          WRITE (IP,24) (SUM(J), J=JJ,JJJ)
62          WRITE (IP,25) (STDV(J), J=JJ,JJJ)
63        280 JJ = JJ + 15
64          IF (NEQ.EQ.0) STOP
      C GENERATE R-MATRIX OF CORRELATIONS FOR KTH PREDICTION EQUATI
65        290 DO 500 K=1,NEQ
66          READ (IR,3) NC, (IX(J), J=1,NC)
67          NR = NC - 1
68          IC = IX(NC)
69          DO 300 I=1,NR
70          II = IX(I)
71          DO 300 J=1,NC
72          JJ = IX(J)
73        300 R(I,J) = SS(II,JJ)
      C SOLVE FOR STANDARD-WEIGHTS USING MODIFIED GAUSS-JORDAN PROC
74          DO 400 I=1,NR
75          BB = R(I,I)
76          IF (ABS(BB).LT.0.0001) BB = 1.0
77          DO 310 J=1,NC
78        310 R(I,J) = R(I,J) / BB
79          DO 400 L=1,NR
80          IF (I.EQ.L) GO TO 400
81          AA = -R(L,I)
82          DO 340 J=1,NC
83        340 R(L,J) = R(L,J) + R(I,J) * AA
84        400 CONTINUE
85          RANK(K) = 0.0
86          DO 410 J=1,NR
87        410 RANK(K) = RANK(K) + R(J,J)
88          WRITE (IP,31) K, IC, RANK(K)
      C CONVERT STD-WEIGHTS TO B-WEIGHTS, CALCULATE CONSTANT, RSQ,
89          RSQ(K) = 0.0
90          CONST = 0.0
91          DO 430 I=1,NR
92          II = IX(I)
93          RSQ(K) = RSQ(K) + R(I,NC) * SS(II,IC)
94          B(I) = R(I,NC) * STDV(IC) / STDV(II)
95          WRITE (IP,32) II, B(I), R(I,NC)
96        430 CONST = CONST + B(I) * SUM(II)
```

```
UMS OF SQUARES, THEN MEANS AND STDVS, THEN CORRELATIONS.      UU 0600
                                                              UU 0610
 I = 1,NV                                                     UU 0620
 J=1,NV                                                       UU 0630
  = SS(I,J) - (SUM(I) * SUM(J) / FN)                          UU 0640
 J=1,NV                                                       UU 0650
 = SUM(J) / FN + C(J)                                         UU 0660
  = SQRT (SS(J,J) / FN) + 0.0000001                           UU 0670
 I=1,NV                                                       UU 0680
 J=1,NV                                                       UU 0690
  = SS(I,J) / (FN * STDV(I) * STDV(J))                        UU 0700
LATIONS, MEANS, AND STANDARD DEVIATIONS IN MATRIX FORM.       UU 0710
                                                              UU 0720
V - 1) / 15 + 1                                               UU 0730
 M=1,MM                                                       UU 0740
  * 15                                                        UU 0750
IP,22) (J, J=JJ,JJJ)                                          UU 0760
JJ + 14                                                       UU 0770
.LE.JJJ) JJJ = NV                                             UU 0780
 I=1,NV                                                       UU 0790
IP,23) I, (SS(I,J), J=JJ,JJJ)                                 UU 0800
IP,24) (SUM(J), J=JJ,JJJ)                                     UU 0810
IP,25) (STDV(J), J=JJ,JJJ)                                    UU 0820
J + 15                                                        UU 0830
.EQ.0) STOP                                                   UU 0840
-MATRIX OF CORRELATIONS FOR KTH PREDICTION EQUATION.          UU 0850
 K=1,NEQ                                                      UU 0860
R,3) NC, (IX(J), J=1,NC)                                      UU 0870
 - 1                                                          UU 0880
(NC)                                                          UU 0890
 I=1,NR                                                       UU 0900
(I)                                                           UU 0910
 J=1,NC                                                       UU 0920
(J)                                                           UU 0930
 = SS(II,JJ)                                                  UU 0940
STANDARD-WEIGHTS USING MODIFIED GAUSS-JORDAN PROCEDURE.       UU 0950
 I=1,NR                                                       UU 0960
(I,I)                                                         UU 0970
(BB).LT.0.0001) BB = 1.0                                      UU 0980
 J=1,NC                                                       UU 0990
 = R(I,J) / BB                                                UU 1000
 L=1,NR                                                       UU 1010
Q.L) GO TO 400                                                UU 1020
(L,I)                                                         UU 1030
 J=1,NC                                                       UU 1040
 = R(L,J) + R(I,J) * AA                                       UU 1050
E                                                             UU 1060
 = 0.0                                                        UU 1070
 J=1,NR                                                       UU 1080
 = RANK(K) + R(J,J)                                           UU 1090
IP,31) K, IC, RANK(K)                                         UU 1100
-WEIGHTS TO B-WEIGHTS, CALCULATE CONSTANT, RSQ, MULTIPLE-R.   UU 1110
 = 0.0                                                        UU 1120
 0.0                                                          UU 1130
 I=1,NR                                                       UU 1140
(I)                                                           UU 1150
 = RSQ(K) + R(I,NC) * SS(II,IC)                               UU 1160
 R(I,NC) * STDV(IC) / STDV(II)                                UU 1170
IP,32) II, B(I), R(I,NC)                                      UU 1180
 CONST + B(I) * SUM(II)                                       UU 1190
```

```
      CONST = SUM(IC) - CONST
      RMULT = SQRT (RSQ(K))
  500 WRITE (IP,33) CONST, RMULT, RSQ(K)
      IF (NF.EQ.0) STOP
      DO 600 K=1,NF
      READ (IR,4) IRSQF, IRSQR, IDF1, IDF2, LABEL
      IF (IDF1.NE.0) GO TO 590
      IDF1 = RANK(IRSQF) - RANK(IRSQR) + 0.4999
      IDF2 = FN - RANK(IRSQF) - 0.5001
  590 F = (RSQ(IRSQF) - RSQ(IRSQR)) *IDF2 /((1.0 - RSQ(IRSQF))
  600 WRITE (IP,51) IRSQF, IRSQR, F, IDF1, IDF2, LABEL
      STOP
      END
```

```
NST                                                              UU 1200
)                                                                UU 1210
 RMULT, RSQ(K)                                                   UU 1220
                                                                 UU 1230
                                                                 UU 1240
RSQR, IDF1, IDF2, LABEL                                          UU 1250
 590                                                             UU 1260
 RANK(IRSQR) + C.4999                                            UU 1270
QF) - 0.5001                                                     UU 1280
(IRSQR)) *IDF2 /((1.0 - RSQ(IRSQF)) *IDF1)                       UU 1290
 IRSQR, F, IDF1, IDF2, LABEL                                     UU 1300
                                                                 UU 1310
                                                                 UU 1320
```

TITLE: X(1-4) FROM ROSCOE P278. X(5)=X(4)+5. X(5)=X(6)+X(7). X(9)=5.

VAR.NO.	1	2	3	4	5	6	7	8
1	1.0000	-0.5000	-0.5000	-0.5000	-0.5000	-0.4264	-0.3571	-0.8216
2	-0.5000	1.0000	-0.5000	0.0000	0.0000	-0.1066	0.0714	0.0000
3	-0.5000	-0.5000	1.0000	0.5000	0.5000	0.5330	0.2857	0.8216
4	-0.5000	0.0000	0.5000	1.0000	1.0000	0.6396	0.8571	0.7912
5	-0.5000	0.0000	0.5000	1.0000	1.0000	0.6396	0.8571	0.7912
6	-0.4264	-0.1066	0.5330	0.6396	0.6396	1.0000	0.1523	0.7006
7	-0.3571	0.0714	0.2857	0.8571	0.8571	0.1523	1.0000	0.5477
8	-0.8216	0.0000	0.8216	0.7912	0.7912	0.7006	0.5477	1.0000
9	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
10	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
MEANS	0.333	0.333	0.333	2.000	7.000	3.111	3.889	5.000
STDV.	0.471	0.471	0.471	1.414	1.414	0.737	1.100	2.582

REGRESSION EQUATION NO. 1 CRITERION IS VAR. NO. 8

VAR.NO.	B-WEIGHT	Z-WEIGHT
4	1.0598	0.5805
5	0.0000	0.0000
6	1.1538	0.3294
7	-0.0000	-0.0000
9	0.0000	0.0000
10	0.0000	0.0000
CONSTANT =	-0.7094	
MULTPL-R =	0.8307	
R-SQUARE =	0.6900	

REGRESSION EQUATION NO. 2 CRITERION IS VAR. NO. 8

VAR.NO.	B-WEIGHT	Z-WEIGHT
9	0.0000	0.0000
7	1.0598	0.4515
6	2.2137	0.6319
5	-0.0000	-0.0000
4	-0.0000	-0.0000
CONSTANT =	-6.0085	
MULTPL-R =	0.8307	
R-SQUARE =	0.6900	

(4)+5. X(5)=X(6)+X(7). X(9)=5. X(10)=BLANK FIELD. N = 9

```
   5       6       7       8       9       10      11      12      13      14      15
```

0.5000 -0.4264 -0.3571 -0.8216 0.0000 0.0000

0.0000 -0.1066 0.0714 0.0000 0.0000 0.0000

0.5000 0.5330 0.2857 0.8216 0.0000 0.0000

1.0000 0.6396 0.8571 0.7912 0.0000 0.0000

1.0000 0.6396 0.8571 0.7912 0.0000 0.0000

0.6396 1.0000 0.1523 0.7006 0.0000 0.0000

0.8571 0.1523 1.0000 0.5477 0.0000 0.0000

0.7912 0.7006 0.5477 1.0000 0.0000 0.0000

0.0000 0.0000 0.0000 0.0000 0.0000 0.0000

0.0000 0.0000 0.0000 0.0000 0.0000 0.0000

 7.000   3.111   3.889   5.000   5.000   0.000

 1.414   0.737   1.100   2.582   0.000   0.000

RION IS VAR. NO. 8 RANK = 2.

RION IS VAR. NO. 8 RANK = 2.

```
**********************************************************************
    REGRESSION EQUATION NO.  3          CRITERION IS VAR. NO.  8
**********************************************************************
          VAR.NO.      B-WEIGHT      Z-WEIGHT
                5        1.0598        0.5805
                6        1.1538        0.3294
        CONSTANT =       -6.0085

        MULTPL-R =        0.8307
        R-SQUARE =        0.6900
**********************************************************************
**********************************************************************
    REGRESSION EQUATION NO.  4          CRITERION IS VAR. NO.  4
**********************************************************************
          VAR.NO.      B-WEIGHT      Z-WEIGHT
                1       -2.0000       -0.6667
                2       -1.0000       -0.3333
        CONSTANT =        3.0000

        MULTPL-R =        0.5774
        R-SQUARE =        0.3333
**********************************************************************
**********************************************************************
    REGRESSION EQUATION NO.  5          CRITERION IS VAR. NO.  4
**********************************************************************
          VAR.NO.      B-WEIGHT      Z-WEIGHT
                1       -2.0000       -0.6667
                2       -1.0000       -0.3333
                3        0.0000        0.0000
        CONSTANT =        3.0000

        MULTPL-R =        0.5774
        R-SQUARE =        0.3333
**********************************************************************
    FULL MODEL  5     RES. MODEL 50     F =     1.500     DF =  2,     6
**********************************************************************
```

****
ION IS VAR. NO. 8 RANK = 2.
****

****
****
ION IS VAR. NO. 4 RANK = 2.
****

****
****
ION IS VAR. NO. 4 RANK = 2.
****

****
.500 DF = 2, 6 TEST OF HYPOTHESIS OF EQUAL MEANS
****

```
COMPILE TIME=     5.26 SEC,EXECUTION TIME=     5.88 SEC,OBJECT CODE=

          $STOP
```

```
=C,OBJECT CODE=     6672 BYTES,ARRAY AREA=    23860 BYTES,UNUSED=    69468 BYTES
```